

Récupération de chaleur sous toiture

La récupération de chaleur sous toiture consiste à capter la chaleur accumulée sous le toit, provenant du soleil et de l'intérieur du bâtiment. Cette chaleur est ensuite réutilisée pour chauffer l'air intérieur, réduisant ainsi les besoins en chauffage.

 Difficulté Moyen

 Durée 68 heure(s)

 Catégories Énergie, Mobilier, Recyclage & Upcycling

 Coût 100 EUR (€)

Sommaire

[Étape 1 - évaluation du besoin](#)

[Étape 2 - Cahier des Charges et diagramme SYML](#)

[Étape 3 - Arduino](#)

[Étape 4 - SolidWorks simulation](#)

[Étape 5 - Sketchup](#)

[Étape 6 - Assemblage](#)

[Commentaires](#)

Matériaux

1 carte Arduino ; 2 ventilateurs ; 1 shield ; Câbles groove ; 2 capteurs de température ; 1 écran LCD ; Mousse expansive ; Polyuréthane ; 1 poulie

Outils

MindView, SolidWorks, SketchUp, Arduino, Excell, scies, mètre, cutter, crayon gris, règle, économie,

Étape 1 - évaluation du besoin

Tout d'abord pour bien commencer il faut évaluer les besoin de ce projet.

Premièrement il faut faire des recherches pour trouver tous les matériaux dont tu auras besoin pour mener à bien ce projet.

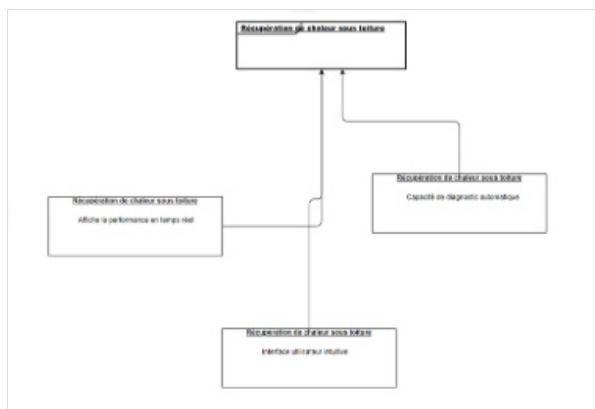
Tu auras besoin : d'un ordinateur pour utiliser les logiciels GANTT, SolidWorks, SketchUp, Arduino et Excell. Une carte Arduino, 2 ventilateurs, 1 shield, des câbles groove et 2 capteurs de température et un écran LCD.

Compte une centaine d'euro pour le matériel sans compter les licences des logiciels et l'ordinateur.

Étape 2 - Cahier des Charges et diagramme SYML

Il faut réaliser les diagrammes de SysML : cas d'utilisation, exigences et block interne.

- Le diagramme de cas d'utilisation en SysML sert à représenter les fonctions ou services qu'un système offre à ses utilisateurs. Il faut que tu identifies les interactions entre les différents acteurs et le système.
- Le diagramme d'exigences en SysML sert à représenter les besoins et contraintes auxquels le système doit répondre. Il faut tracer, organiser et relier les exigences entre elles.
- Le diagramme de blocs internes en SysML sert à représenter la structure interne d'un bloc, en montrant ses composants et leurs interactions. Il permet de visualiser l'architecture interne d'un système. Il faut mettre en évidence les différents éléments qui constituent le système.



Il faut réaliser le cahier des charges du projet. Commence par définir le contexte et les objectifs du projet. Décris ensuite les fonctionnalités attendues, puis précise les contraintes à respecter. Indique les critères de performance, et termine par la liste des livrables attendus.

Étape 3 - Arduino

Tout d'abord tu va faire un programme Arduino ou tu demandera à ton écran de t'afficher la température détecter sur les DEUX capteur. Ensuite tu lui demande d'activer les ventilateurs uniquement quand la température du capteur 1 dépasse 25°C et de ce désactiver quand la température est inférieur.

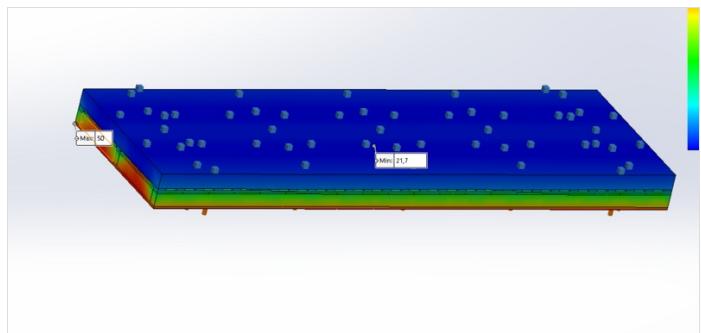
Pour finir tu rajoute une commande pour activer et désactiver quand tu veux le programme.

```
code_mieux.ino
1 #include <DHT.h>
2 #include "rgb_lcd.h" // Bibliothèque pour l'écran LCD RGB
3 #include <Servo.h>
4
5 // Définir les pins des capteurs DHT11
6 #define DHTPIN1 2 // Capteur 1
7 #define DHTPIN2 3 // Capteur 2
8
9 // Définir le type de capteur DHT11
10 #define DHTTYPE DHT11
11
12 // Initialisation des capteurs DHT11
13 DHT dht1(DHTPIN1, DHTTYPE);
14 DHT dht2(DHTPIN2, DHTTYPE);
15
16 // Initialisation de l'écran LCD RGB
17 rgb_lcd lcd;
18
19 // Initialisation du servo moteur
20 Servo servo;
21 int servoPin = 8; // Servo sur D8
22
23 void setup() {
24   Serial.begin(9600);
25
26   // Initialisation de l'écran LCD RGB
27   lcd.begin(16, 2);
28   lcd.setRGB(0, 255, 0); // Par défaut, mettre le rétroéclairage à vert
29
30   // Initialisation des capteurs DHT
31   dht1.begin();
32   dht2.begin();
33
34   // Initialisation du servo
35   servo.attach(servoPin);
36   servo.write(0); // Position initiale à 0°
37
38   // Afficher un message d'accueil
39   lcd.setCursor(0, 0);
40   lcd.print("System Ready");
41   delay(2000);
42   lcd.clear(); // Nettoyer après le message d'accueil
43 }
44
45 void loop() {
46   // Lire les températures des capteurs
47   float temp1 = dht1.readTemperature();
```

```
48
49
50   // Vérifier si les lectures de température sont valides
51   if (isnan(temp1) || isnan(temp2)) {
52     lcd.setCursor(0, 0);
53     lcd.print("Erreur capteurs");
54     Serial.println("Erreur de lecture des capteurs !");
55     delay(2000);
56     return;
57   }
58
59   // Mettre à jour uniquement les valeurs sans réinitialiser l'écran
60   lcd.setCursor(0, 0);
61   lcd.print("Temp 1: ");
62   lcd.print(temp1, 1);
63   lcd.print(" C ");
64
65   lcd.setCursor(0, 1);
66   lcd.print("Temp 2: ");
67   lcd.print(temp2, 1);
68   lcd.print(" C ");
69
70   Serial.print("Temp1: "); Serial.print(temp1); Serial.println(" C");
71   Serial.print("Temp2: "); Serial.print(temp2); Serial.println(" C");
72
73   // Si la température du capteur 1 dépasse 25°C, activer le servo et rester en rouge
74   if (temp1 > 25) {
75     lcd.setRGB(255, 0, 0); // Changer la couleur en rouge
76     servo.write(90); // Activer le servo (90°)
77   } else {
78     servo.write(0); // Désactiver le servo (0°)
79     lcd.setRGB(0, 255, 0); // Remettre la couleur en vert uniquement si le servo est désactivé
80   }
81
82   delay(2000); // Pause avant la prochaine lecture
```

Étape 4 - SolidWorks simulation

Pour commencer tu va crée ta maquette avec tes mesures que tu auras relever sur un modèle réel ou inventer et grâce a ça tu pourra commencer a faire la simulation thermique, pour réaliser cela tu devras aller dans le menu simulation ce qui te permettras d'appliquer les différents paramètres que tu as besoin, la radiation et le flux de chaleur sont des paramètres essentiel pour une bonne simulation simple mais d'autre paramètre peuvent être utiliser.



Étape 5 - Sketchup

Pour commencer, tu vas sélectionner les outils dont tu auras besoin pour réaliser ta construction.

Tout d'abord tu vas prendre ton plan de cadastre que tu trouves sur Google, lorsque tu vas sur Geoportail. Tu l'importe ensuite sur Sketchup, pour le mettre comme "base de support".

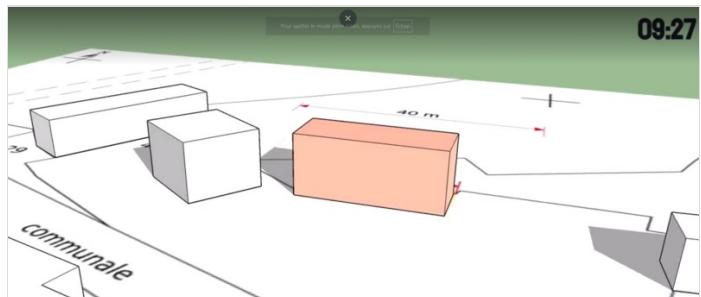
Tu sélectionneras ensuite, les ombres qui est un des outils dont tu auras besoin, pour réaliser ton diagramme solaire.

Tu placeras ensuite ton bâtiment, ainsi que son environnement (mesure réel).

Dans l'outil Ombre, tu changeras les jours, mois, et les heures en fonction de chaque solstices (saisons), pour pouvoir visualiser les phases d'ensoleillement.

Par la suite, tu vas dans Affichage, puis Animation pour créer une Scène, tu peux bouger ta construction, en fonction de l'angle que tu veux donner à ta vidéo.

Pour finir, tu vas dans Ficher, et Exporter, puis Animation et Séquence Vidéo pour créer une vidéo.



Étape 6 - Assemblage

Pour commencer il faut faire l'état des lieux pour évaluer tout le matériel à acheter

Après avoir fait ça il faut faire les mesures après avoir fait les meures il faut faire la découpe et l'installation après avoir fait installation tu vas combler le trou à l'aide d'un mousse expansive

Après ça tu peux passer à la phase de tests en installant ton système que tu as préalablement monter et coder.

